Support Vector Machine and its Appliactions

G P S Raghava

Institute of Microbial Technology

Sector 39A, Chandigrah, India

Email: Raghava@imtech.res.in

Web: http://www.imtech.res.in/raghava/ http://crdd.osdd.net/

Why Machine Learning ?

- Similarity based methods
- Linear seperations
- Statistical methods (static)
- Unable to handle non-linear data

Supervised & Unsupervised

- Learn an unknown function f(X) = Y, where X is an input example and Y is the desired output.
- **Supervised learning** implies we are given a **training set** of (X, Y) pairs by a "teacher"
- **Unsupervised learning** means we are only given the Xs and some (ultimate) feedback function on our performance.

Concept learning or classification

- **Given a set of examples of some concept/class/category, determine if a given example is an instance of the concept or not**
- **If it is an instance, we call it a positive example**
- **If it is not, it is called a negative exampl**
- **Or we can make a probabilistic prediction (e.g., using a Bayes net**)

Supervised concept learning

- **Given a training set of positive and negative examples of a concept**

- **Construct a description that will accurately classify whether future examples are positive or negative**

- **That is, learn some good estimate of function f given a training set {(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)} where each y_i is either + (positive) or - (negative), or a probability distribution over +/-**

Major Machine Learning Technoques

- **Artificial Neural Networks**
- **Hidden Markov Model**
- **Nearest Neighbur Methods**
- **Support Vector Machines**

Introduction to Neural Networks

- **Neural network:** *information processing paradigm inspired by biological nervous systems, such as our brain*
- **Structure:** large number of highly interconnected processing elements (*neurons*) working together
- Like people, they learn *from experience* (by example)
- Neural networks are configured for a specific application

Neural networks to the rescue

- Neural networks are configured for a specific application, such as pattern recognition or data classification, through a learning process
- In a biological system, learning involves adjustments to the synaptic connections between neurons

➔ same for artificial neural networks (ANNs)

Where can neural network systems help

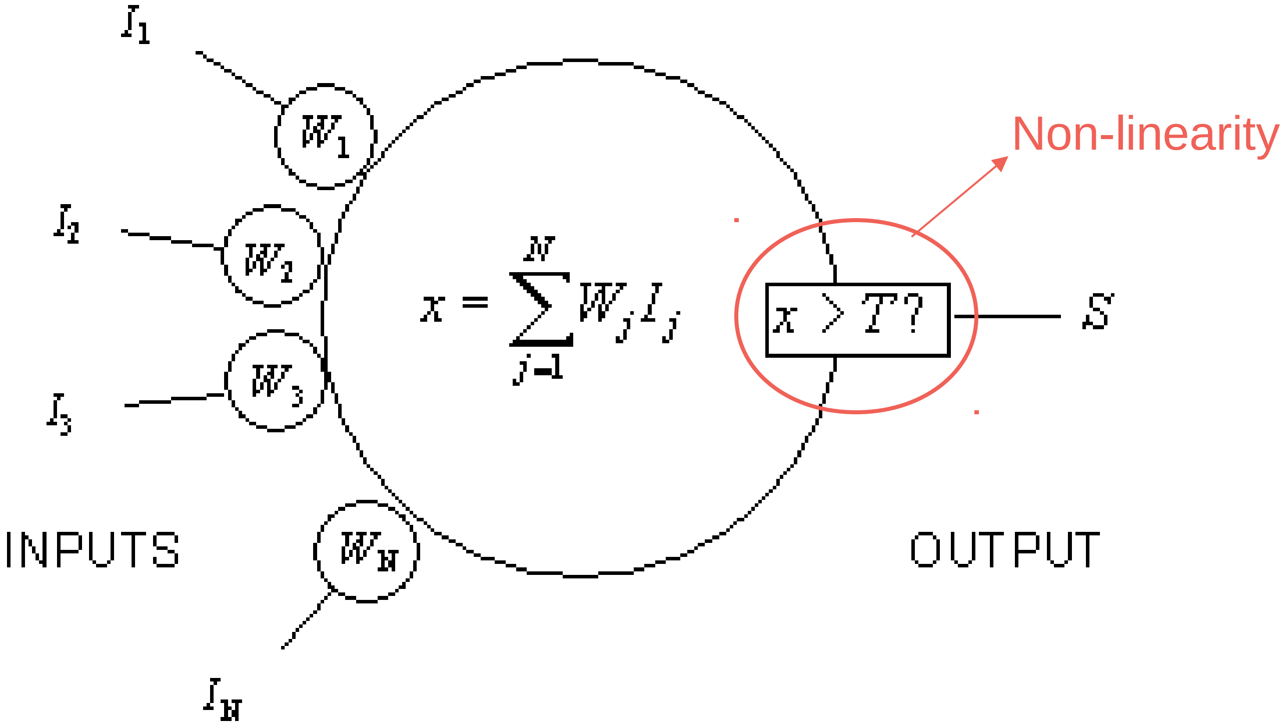
- when we can't formulate an algorithmic solution.
- when we ***can*** get lots of examples of the behavior we require.

‘learning from experience’

- when we need to pick out the structure from existing data.

Mathematical representation

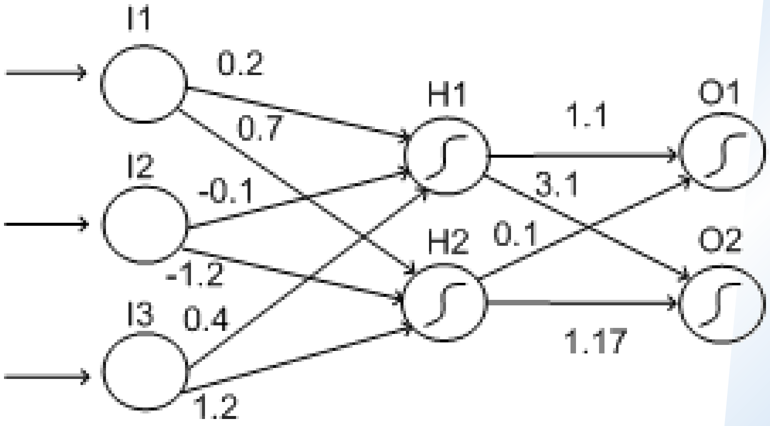
The neuron calculates a weighted sum of inputs and compares it to a threshold. If the sum is higher than the threshold, the output is set to 1, otherwise to -1.



Non-linearity

Artificial Neural Networks

- Layers of nodes
  - Input is transformed into numbers
  - Weighted averages are fed into nodes
- High or low numbers come out of nodes
  - A Threshold function determines whether high or low
- Output nodes will “fire” or not
  - Determines classification
    - For an example



Markov Chains

and hidden Markov models

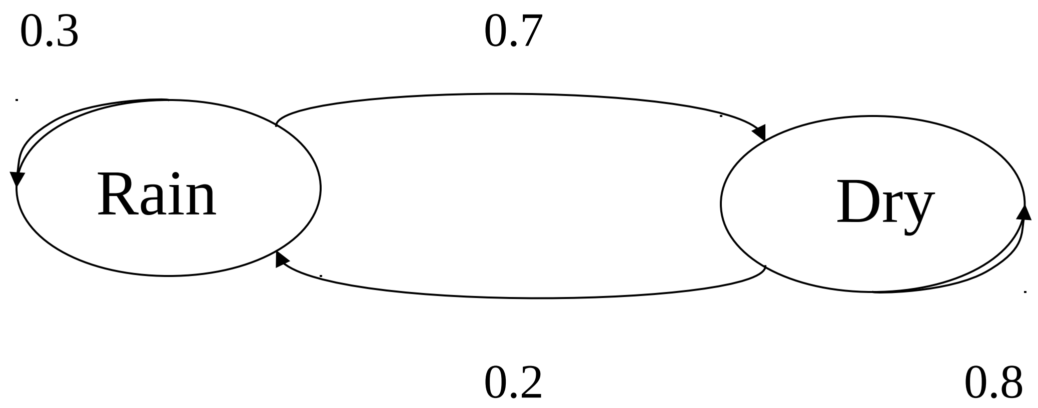
A widely used machine learning approach: Markov models

- Markov chain models (1st order, higher order and inhomogeneous models; parameter estimation; classification)

- Interpolated Markov models (and back-off models)

- Hidden Markov models (forward, backward and Baum-Welch algorithms; model topologies; applications to gene finding and protein family modeling

Example of Markov Model



- Two states : ‘Rain’ and ‘Dry’.
- Transition probabilities: P(‘Rain’|‘Rain’)=0.3 , P(‘Dry’|‘Rain’)=0.7 , P(‘Rain’|‘Dry’)=0.2, P(‘Dry’|‘Dry’)=0.8
- Initial probabilities: say P(‘Rain’)=0.4 , P(‘Dry’)=0.6 .

k Nearest-Neighbors Problem

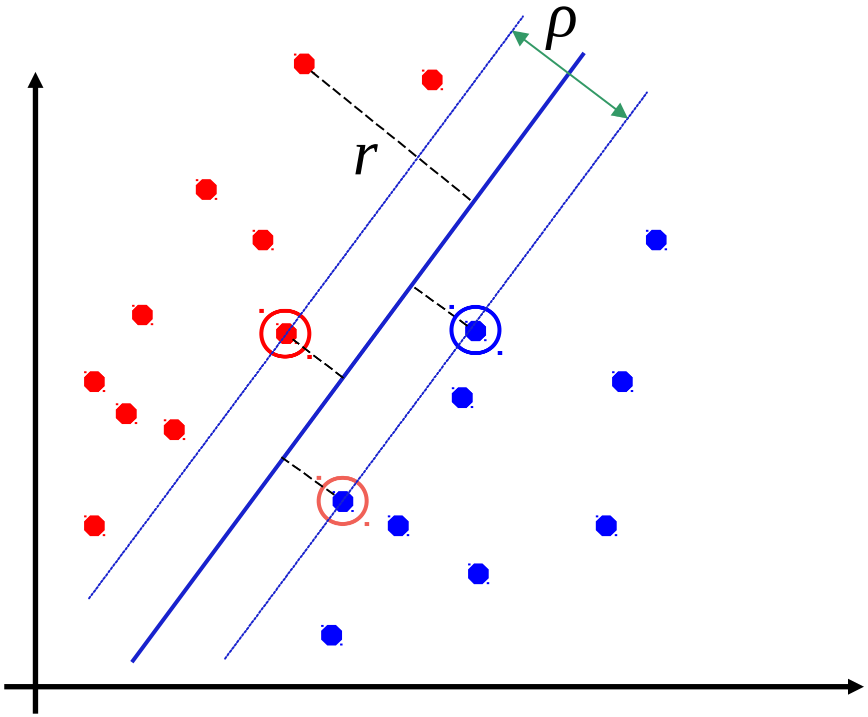
- Example based learning
- Weight for examples
- Closest examples for decision
- Time consuming
- Fail in absence of sufficient examples
- Performance depend on closesness

SVM: Support Vector Machine

- Support vector machines (SVM) are a group of supervised learning methods that can be applied to classification or regression. Support vector machines represent an extension to nonlinear models of the generalized portrait algorithm developed by Vladimir Vapnik. The SVM algorithm is based on the statistical learning theory and the Vapnik-Chervonenkis (VC) dimension introduced by Vladimir Vapnik and Alexey Chervonenkis in 1992.

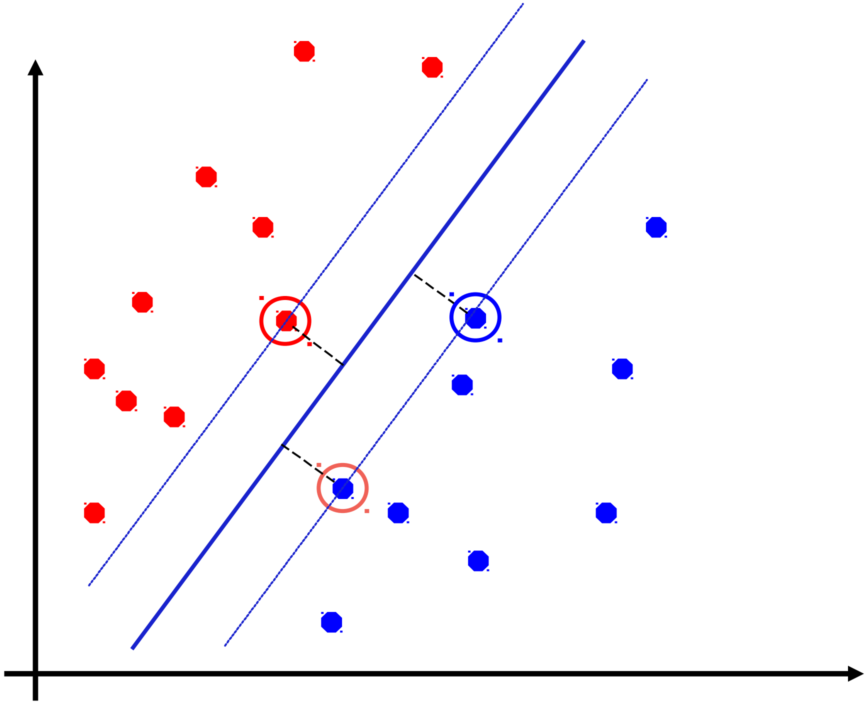
Classification Margin

- Distance from example to the separator is $r = \frac{\mathbf{w}^T\mathbf{x}+b}{\|\mathbf{w}\|}$
- Examples closest to the hyperplane are ***support vectors***.
- ***Margin*** *ρ* of the separator is the width of separation between classes.



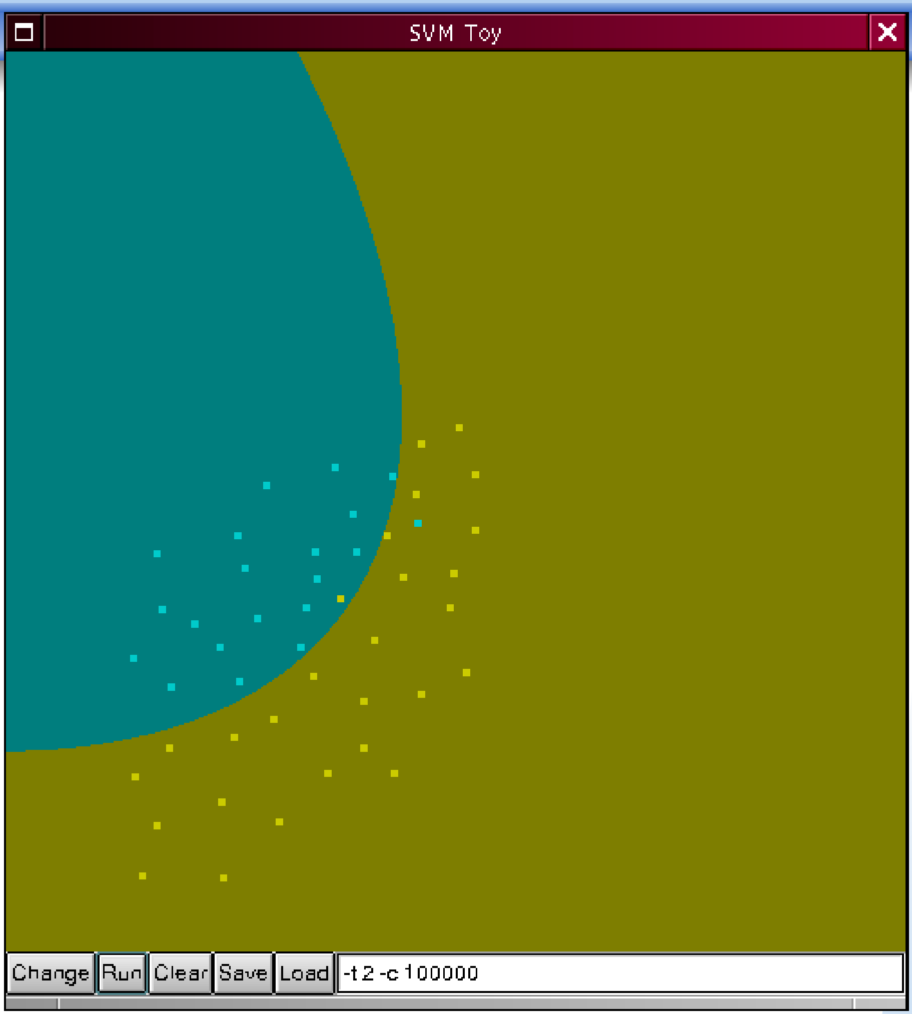
Maximum Margin Classification

- Maximizing the margin is good
- Implies that only support vectors are important;
- other training examples are ignorable.



SVM implementations

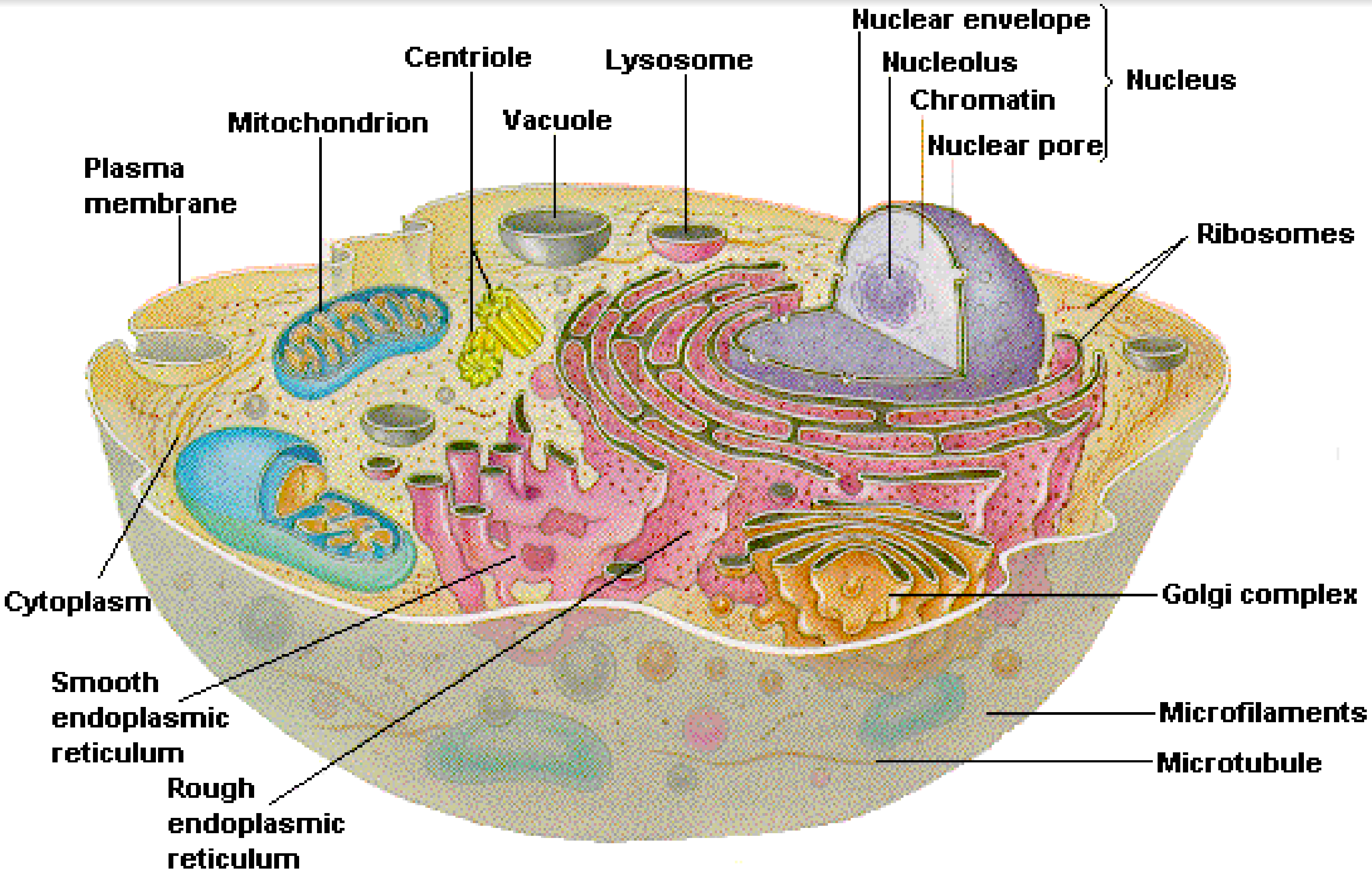
- SVM^{light}
  - Simple text data format
  - Fast, C routines
- bsvm
  - Multiple class.
- LIBSVM
  - GUI: svm-toy
- SMO
  - Less optimization
  - Fast
  - Weka implemented



-t 2 -c 100000

❖**Differences:** available Kernel functions, optimization, multiple class., user interfaces

Subcellular Locations



PREDICTION OF PROTEINS TO BE LOCALIZED IN MITOCHONDRIA (MITPRED)

Mitochondrial Located proteins
(Positive dataset)

```
>Mit1
DRLVRGFYFLLRRMV
SHNTVSQVWFGHRYS
```

Non-mitochondiral Located proteins
(Negative dataset)

```
>Non-Mit1
KNRNTKVGSDRLVRG
WFGHRYSMVHS
```

fasta2sfasta.pl program

```
>Mit1
##DRLVRGFYFLLRRMVSHNTVSQVWFGHR
YS
>Mit2
##RMVKNRNTKVGDRLVRGFYFLLRR
```

```
>Non-Mit1
##KNRNTKVGSDRLVRGWFGHRYSMVHS
>Non-Mit2
##LVRGFYFLLRRMVKNRNSHRVSQ
```

pro2aac.pl program

```
# Amino Acid Composition of Mit proteins
# A , C , D , E , F , G , H , I , K , L , M , N , P , Q , R , S , T , V , W , Y
0.0,0.0,3.3,0.0,10.0,6.7,6.7,0.0,0.0,10.0,3.3,3.3,0.0,3.3,16.7,10.0,3.
3,13.3,3.3,6.7
0.0,0.0,4.2,0.0,8.3,8.3,0.0,0.0,8.3,12.5,4.2,8.3,0.0,0.0,25.0,0.0,4.2,
12.5,0.0,4.2
```

```
# Amino Acid Composition of Non-Mit proteins
# A , C , D , E , F , G , H , I , K , L , M , N , P , Q , R , S , T , V , W , Y
0.0,0.0,3.8,0.0,3.8,11.5,7.7,0.0,7.7,3.8,3.8,7.7,0.0,0.0,15.4,11.5,3.8,1
1.5,3.8,3.8
0.0,0.0,0.0,0.0,8.7,4.3,4.3,0.0,4.3,13.0,4.3,8.7,0.0,4.3,21.7,8.7,0.0,13
.0,0.0,4.3
```

col2svm.pl program

```
+1 1:0.0 2:0.0 3:3.3 4:0.0 5:10.0 6:6.7
7:6.7 8:0.0 9:0.0 10:10.0 11:3.3 12:3.3
13:0.0 14:3.3 15:16.7 16:10.0 17:3.3
18:13.3 19:3.3 20:6.7
+1 1:0.0 2:0.0 3:4.2 4:0.0 5:8.3 6:8.3
7:0.0 8:0.0 9:8.3 10:12.5 11:4.2 12:8.3
13:0.0 14:0.0 15:25.0 16:0.0 17:4.2
18:12.5 19:0.0 20:4.2
```

```
-1 1:0.0 2:0.0 3:3.8 4:0.0 5:3.8 6:11.5
7:7.7 8:0.0 9:7.7 10:3.8 11:3.8 12:7.7
13:0.0 14:0.0 15:15.4 16:11.5 17:3.8
18:11.5 19:3.8 20:3.8
-1 1:0.0 2:0.0 3:0.0 4:0.0 5:8.7 6:4.3
7:4.3 8:0.0 9:4.3 10:13.0 11:4.3 12:8.7
13:0.0 14:4.3 15:21.7 16:8.7 17:0.0
18:13.0 19:0.0 20:4.3
```

SVM-input file

PREDICTION OF PROTEINS TO BE LOCALIZED IN MITOCHONDRIA (MITPRED)

```
+1 1:0.0 2:0.0 3:3.3 4:0.0 5:10.0 6:6.7 7:6.7 8:0.0 9:0.0 10:10.0 11:3.3 12:3.3 13:0.0 14:3.3 15:16.7 16:10.0
17:3.3 18:13.3 19:3.3 20:6.7
+1 1:0.0 2:0.0 3:4.2 4:0.0 5:8.3 6:8.3 7:0.0 8:0.0 9:8.3 10:12.5 11:4.2 12:8.3 13:0.0 14:0.0 15:25.0 16:0.0 17:4.2
18:12.5 19:0.0 20:4.2
-1 1:0.0 2:0.0 3:3.8 4:0.0 5:3.8 6:11.5 7:7.7 8:0.0 9:7.7 10:3.8 11:3.8 12:7.7 13:0.0 14:0.0 15:15.4 16:11.5 17:3.8
18:11.5 19:3.8 20:3.8
-1 1:0.0 2:0.0 3:0.0 4:0.0 5:8.7 6:4.3 7:4.3 8:0.0 9:4.3 10:13.0 11:4.3 12:8.7 13:0.0 14:4.3 15:21.7 16:8.7 17:0.0
18:13.0 19:0.0 20:4.3
```

Training file

```
+1 1:0.0 2:0.0 3:3.3 4:0.0 5:10.0 6:6.7 7:6.7
8:0.0 9:0.0 10:10.0 11:3.3 12:3.3 13:0.0
14:3.3 15:16.7 16:10.0 17:3.3 18:13.3 19:3.3
20:6.7
-1 1:0.0 2:0.0 3:0.0 4:0.0 5:8.7 6:4.3 7:4.3
8:0.0 9:4.3 10:13.0 11:4.3 12:8.7 13:0.0
14:4.3 15:21.7 16:8.7 17:0.0 18:13.0 19:0.0
20:4.3
```

svm_learn trainng file model

Test file

```
+1 1:0.0 2:0.0 3:4.2 4:0.0 5:8.3 6:8.3 7:0.0
8:0.0 9:8.3 10:12.5 11:4.2 12:8.3 13:0.0
14:0.0 15:25.0 16:0.0 17:4.2 18:12.5 19:0.0
20:4.2
-1 1:0.0 2:0.0 3:3.8 4:0.0 5:3.8 6:11.5 7:7.7
8:0.0 9:7.7 10:3.8 11:3.8 12:7.7 13:0.0
14:0.0 15:15.4 16:11.5 17:3.8 18:11.5 19:3.8
20:3.8
```

svm_classify test-file model result

This result file contains a numeric value, using this value we can evaluate the model performance by varying threshold

SVM_light training/testing pattern

- **Output Input (frequency)**
- **0.902** 1:3 2:8 3:6 4:4 5:0 6:0 7:2
- **0.897** 1:3 2:5 3:6 4:7 5:0 6:0 7:2
- **0.545** 1:3 2:7 3:5 4:6 5:0 6:0 7:2
- **0.850** 1:6 2:4 3:6 4:5 5:2 6:0 7:1
- **0.408** 1:6 2:9 3:2 4:4 5:3 6:2 7:1
- **0.019** 1:4 2:8 3:4 4:5 5:1 6:1 7:1
- **0.834** 1:3 2:7 3:2 4:9 5:0 6:1 7:1
- **0.323** 1:3 2:9 3:3 4:6 5:0 6:2 7:1
- **0.862** 1:8 2:2 3:5 4:6 5:4 6:0 7:2
- **0.284** 1:9 2:2 3:3 4:7 5:4 6:0 7:1
- **1.341** 1:5 2:6 3:4 4:6 5:2 6:0 7:1

```
svm_learn train.svm model

svm_classify test model  output
```

Options

-z c   for classification

-z r for regression

-t 0 linear kernel

-t 1 polynomial

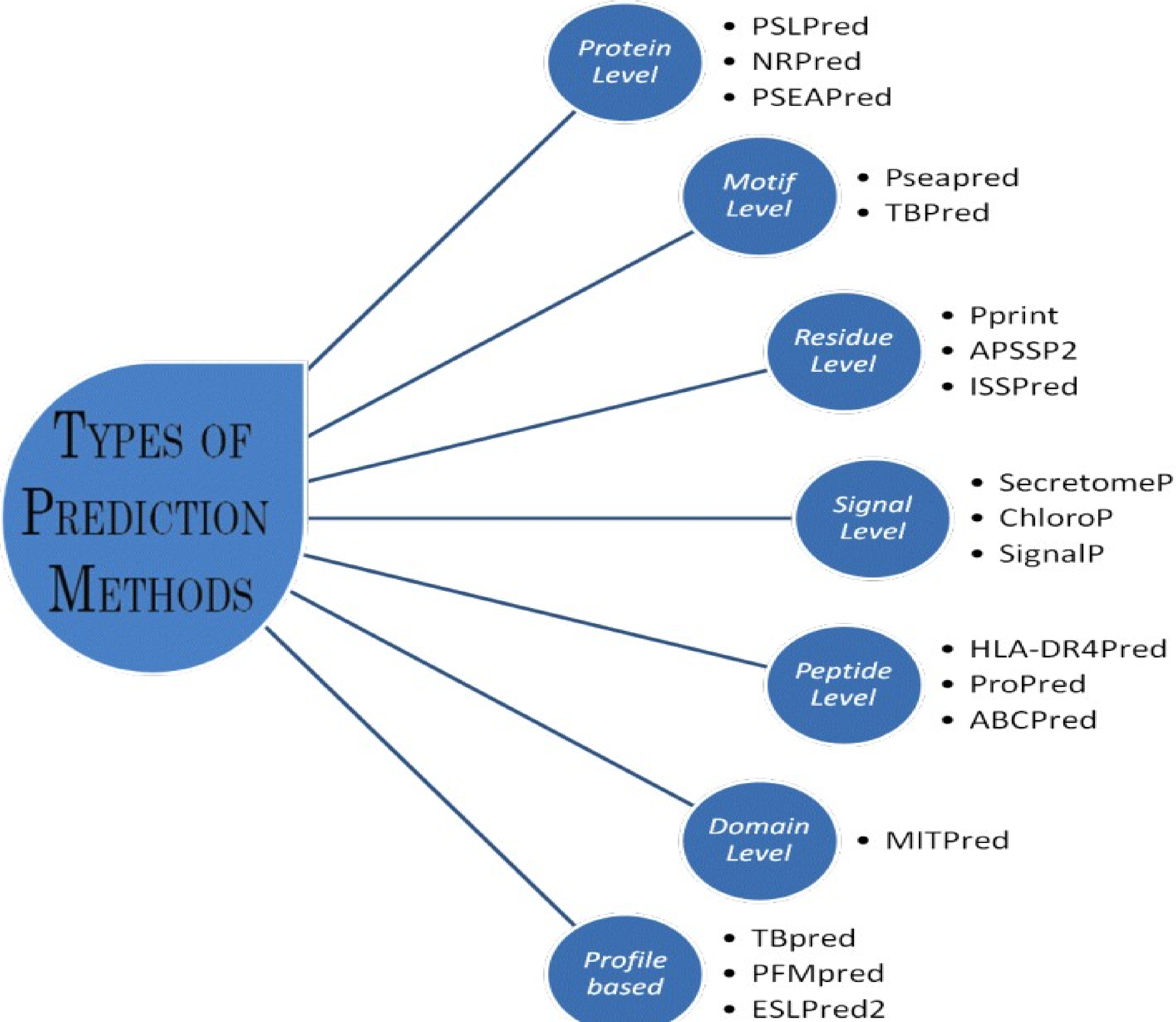
-t 2 RBF

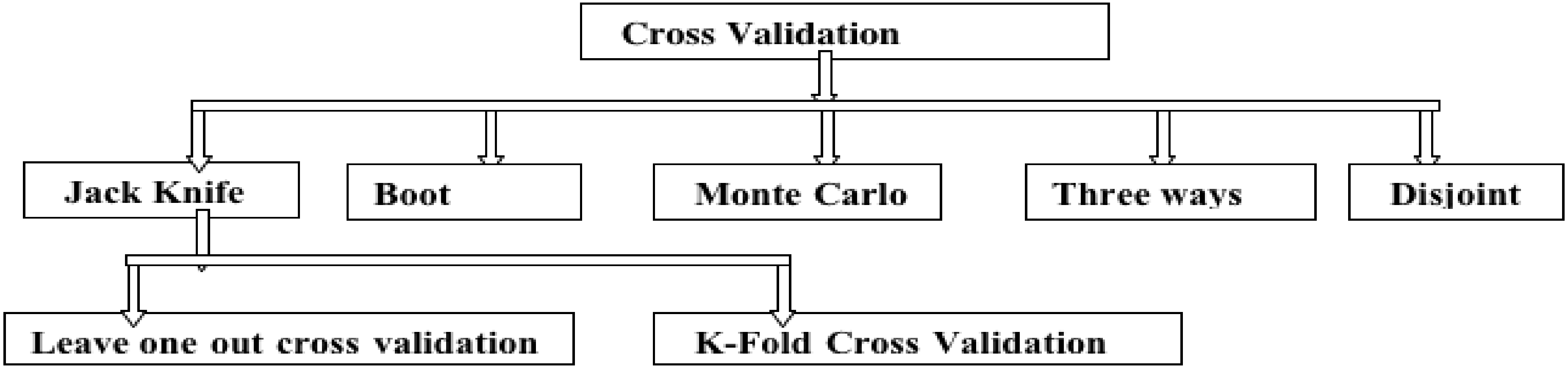
Important Points in Developing New Method

- Importance of problem
- Acceptable dataset
  - Dataset should be large
  - Recently used in any other study
  - Realistic, balance & independent
  - Level of redundancy
- Develop standalone and/or web service
- Cross-validation (Benchmarking)

Important Points in Developing New Method (Cont.)

- **Integrate BLAST with ML techniques**
- **Using PSIBLAST profile**
- **Discover exclusive domian/motif present or absent in proteins.**
- **Features from proteins (fixed length pattern)**
  - **Amino acid composition (split composition)**
  - **Dipeptide composition (higher order)**
  - **Pseudo amino acid composition**
  - **PSSM composition**


Types of Prediction Methods
Protein Level
PSLPred
NRPred
PSEAPred
Motif Level
Pseapred
TBPred
Residue Level
Pprint
APSSP2
ISSPred
Signal Level
SecretomeP
ChloroP
SignalP
Peptide Level
HLA-DR4Pred
ProPred
ABCPred
Domain Level
MITPred
Profile based
TBpred
PFMpred
ESLPred2



Cross Validation

Jack Knife

Boot

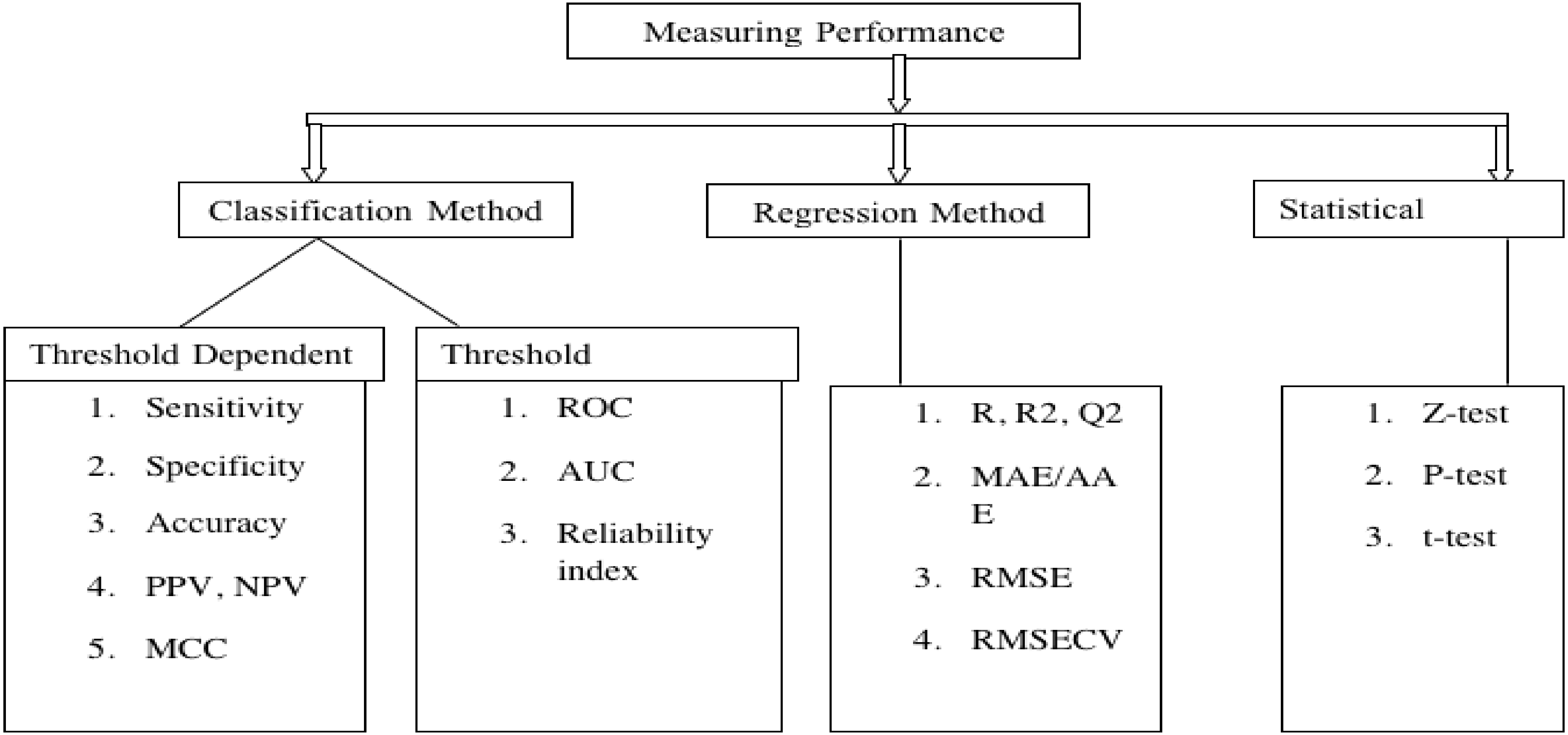
Monte Carlo

Three ways

Disjoint

Leave one out cross validation

K-Fold Cross Validation



Measuring Performance

Classification Method

Regression Method

Statistical

Threshold Dependent

1. Sensitivity
2. Specificity
3. Accuracy
4. PPV, NPV
5. MCC

Threshold

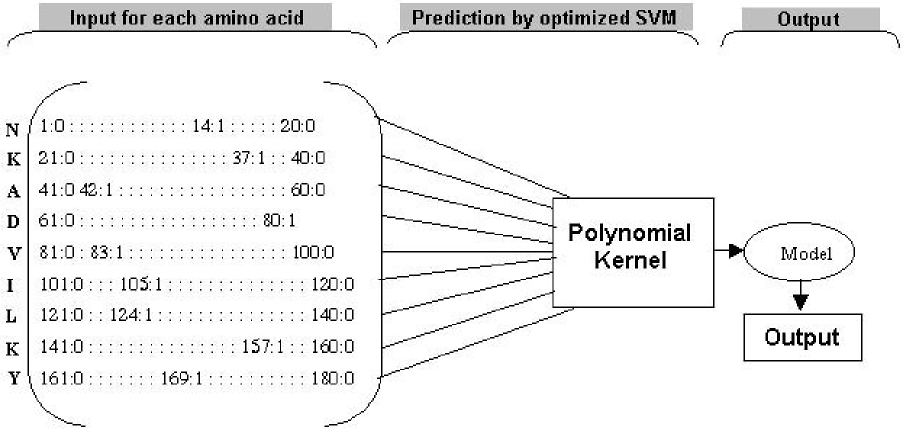
1. ROC
2. AUC
3. Reliability index

1. R, R2, Q2
2. MAE/AA E
3. RMSE
4. RMSECV

1. Z-test
2. P-test
3. t-test

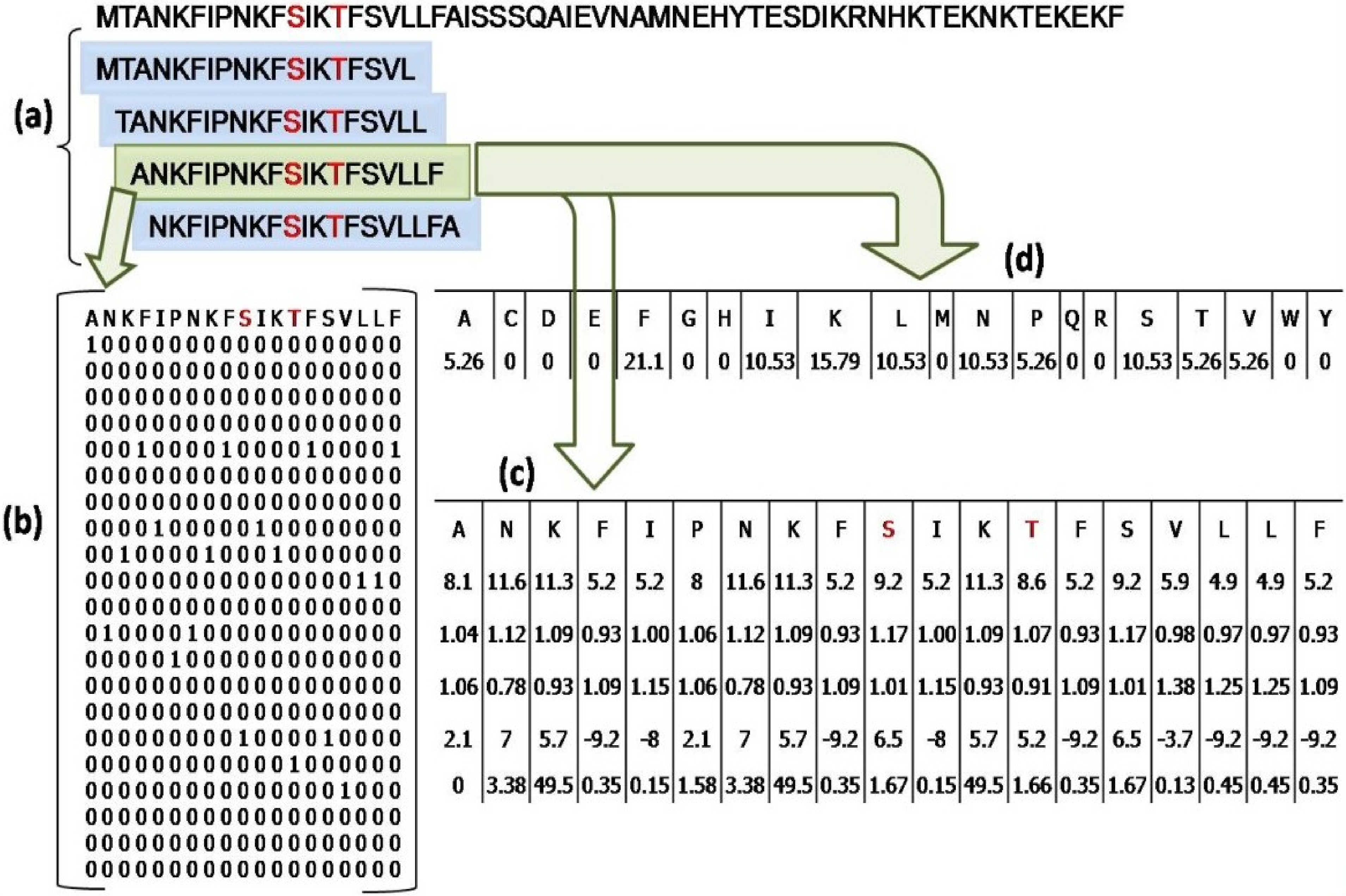
Creation of Pattern

- Fix the length of pattern
  - For example protein (composition)
  - Represent Segment by vector



N 1:0 : : : : : : : : : : : : 14:1 : : : : : 20:0
K 21:0 : : : : : : : : : : : : : : : 37:1 : : 40:0
A 41:0 42:1 : : : : : : : : : : : : : : : : : 60:0
D 61:0 : : : : : : : : : : : : : : : : : : 80:1
V 81:0 : 83:1 : : : : : : : : : : : : : : : : 100:0
I 101:0 : : : 105:1 : : : : : : : : : : : : : : 120:0
L 121:0 : : 124:1 : : : : : : : : : : : : : : : 140:0
K 141:0 : : : : : : : : : : : : : : : 157:1 : : 160:0
Y 161:0 : : : : : : : 169:1 : : : : : : : : : : 180:0

Feature extraction from an antigen primary sequence



MTANKFIPNKFSIKTFSVLLFAISSSQAIEVNAMNEHYTESDIKRNHKTEKNKTEKEKF

(a)

MTANKFIPNKFSIKTFSVL

TANKFIPNKFSIKTFSVLL

ANKFIPNKFSIKTFSVLLF

NKFIPNKFSIKTFSVLLFA

(b)

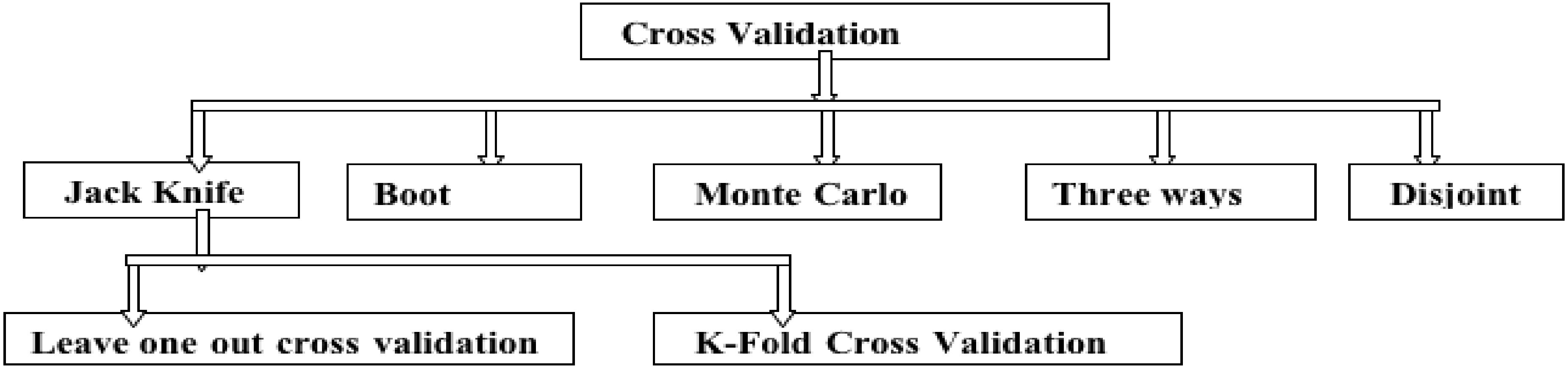
```
ANKFIPNKFSIKTFSVLLF
1000000000000000000
0000000000000000000
0000000000000000000
0000000000000000000
0001000010000100001
0000000000000000000
0000000000000000000
0000100000100000000
0010000100010000000
0000000000000000110
0000000000000000000
0100001000000000000
0000010000000000000
0000000000000000000
0000000000000000000
0000000001000010000
0000000000001000000
0000000000000001000
0000000000000000000
0000000000000000000
```

(d)

A	C	D	E	F	G	H	I	K	L	M	N	P	Q	R	S	T	V	W	Y
5.26	0	0	0	21.1	0	0	10.53	15.79	10.53	0	10.53	5.26	0	0	10.53	5.26	5.26	0	0

(c)

A	N	K	F	I	P	N	K	F	S	I	K	T	F	S	V	L	L	F
8.1	11.6	11.3	5.2	5.2	8	11.6	11.3	5.2	9.2	5.2	11.3	8.6	5.2	9.2	5.9	4.9	4.9	5.2
1.04	1.12	1.09	0.93	1.00	1.06	1.12	1.09	0.93	1.17	1.00	1.09	1.07	0.93	1.17	0.98	0.97	0.97	0.93
1.06	0.78	0.93	1.09	1.15	1.06	0.78	0.93	1.09	1.01	1.15	0.93	0.91	1.09	1.01	1.38	1.25	1.25	1.09
2.1	7	5.7	-9.2	-8	2.1	7	5.7	-9.2	6.5	-8	5.7	5.2	-9.2	6.5	-3.7	-9.2	-9.2	-9.2
0	3.38	49.5	0.35	0.15	1.58	3.38	49.5	0.35	1.67	0.15	49.5	1.66	0.35	1.67	0.13	0.45	0.45	0.35



Cross Validation

Jack Knife

Boot

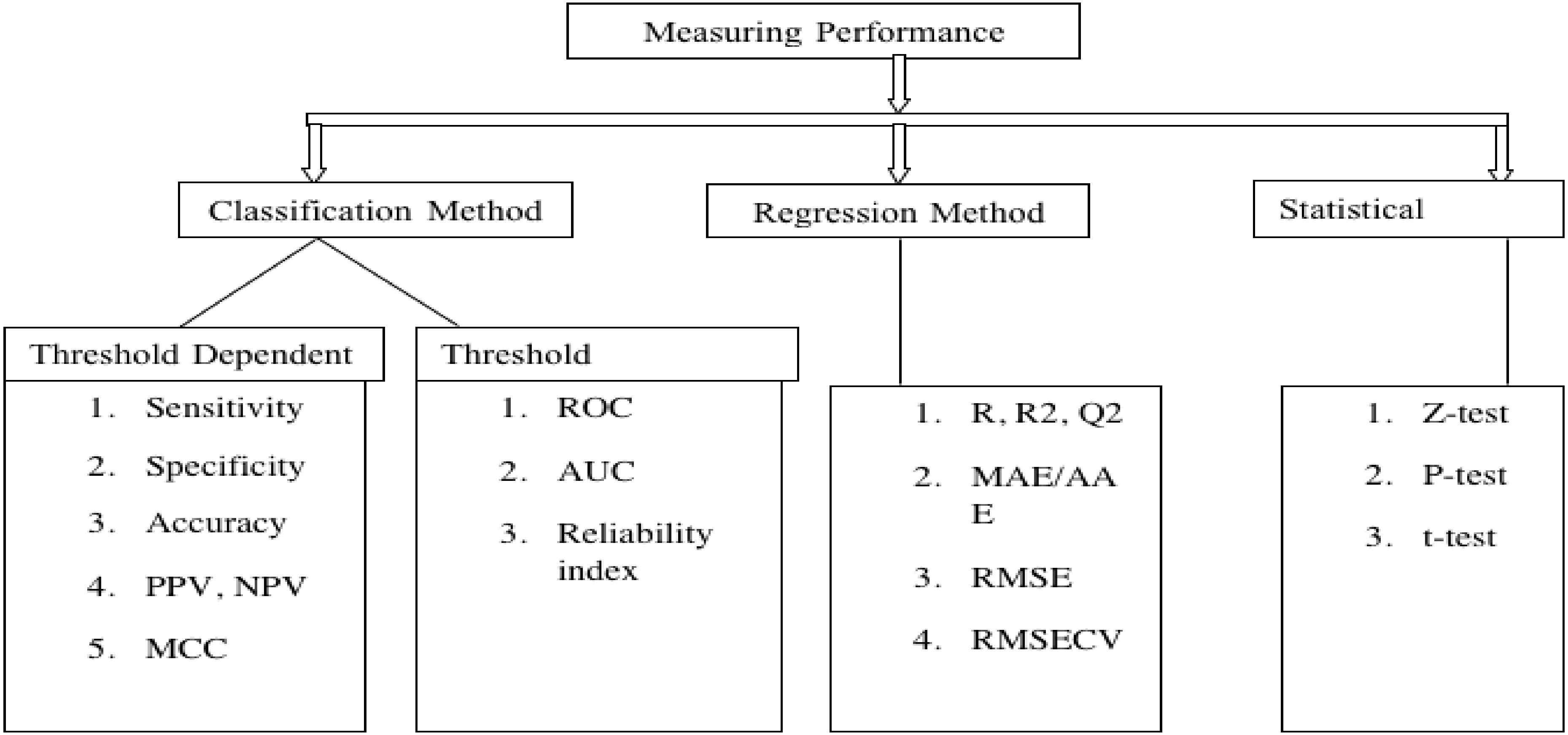
Monte Carlo

Three ways

Disjoint

Leave one out cross validation

K-Fold Cross Validation



Measuring Performance

Classification Method

Regression Method

Statistical

Threshold Dependent

1. Sensitivity
2. Specificity
3. Accuracy
4. PPV, NPV
5. MCC

Threshold

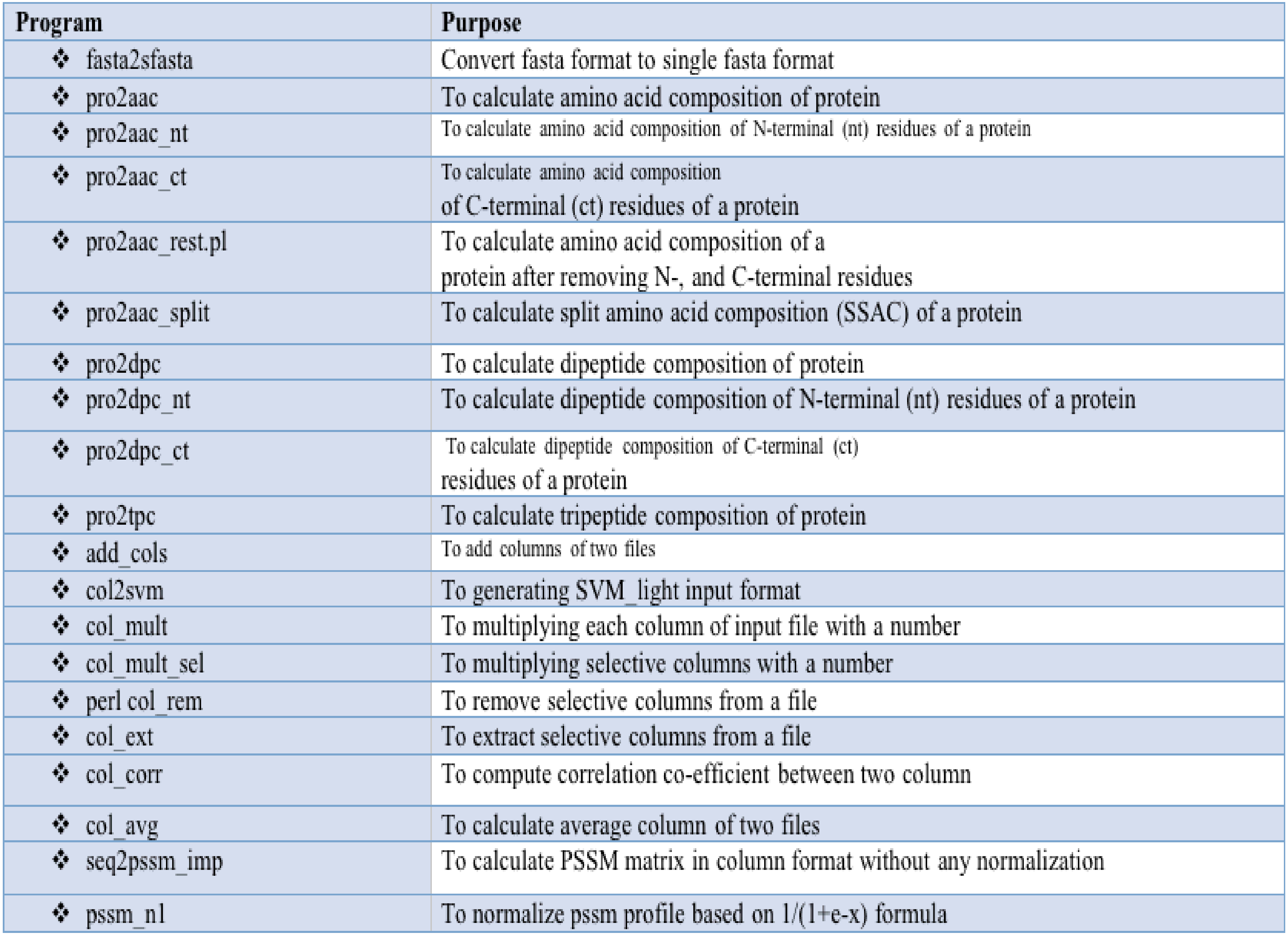
1. ROC
2. AUC
3. Reliability index

1. R, R2, Q2
2. MAE/AA E
3. RMSE
4. RMSECV

1. Z-test
2. P-test
3. t-test

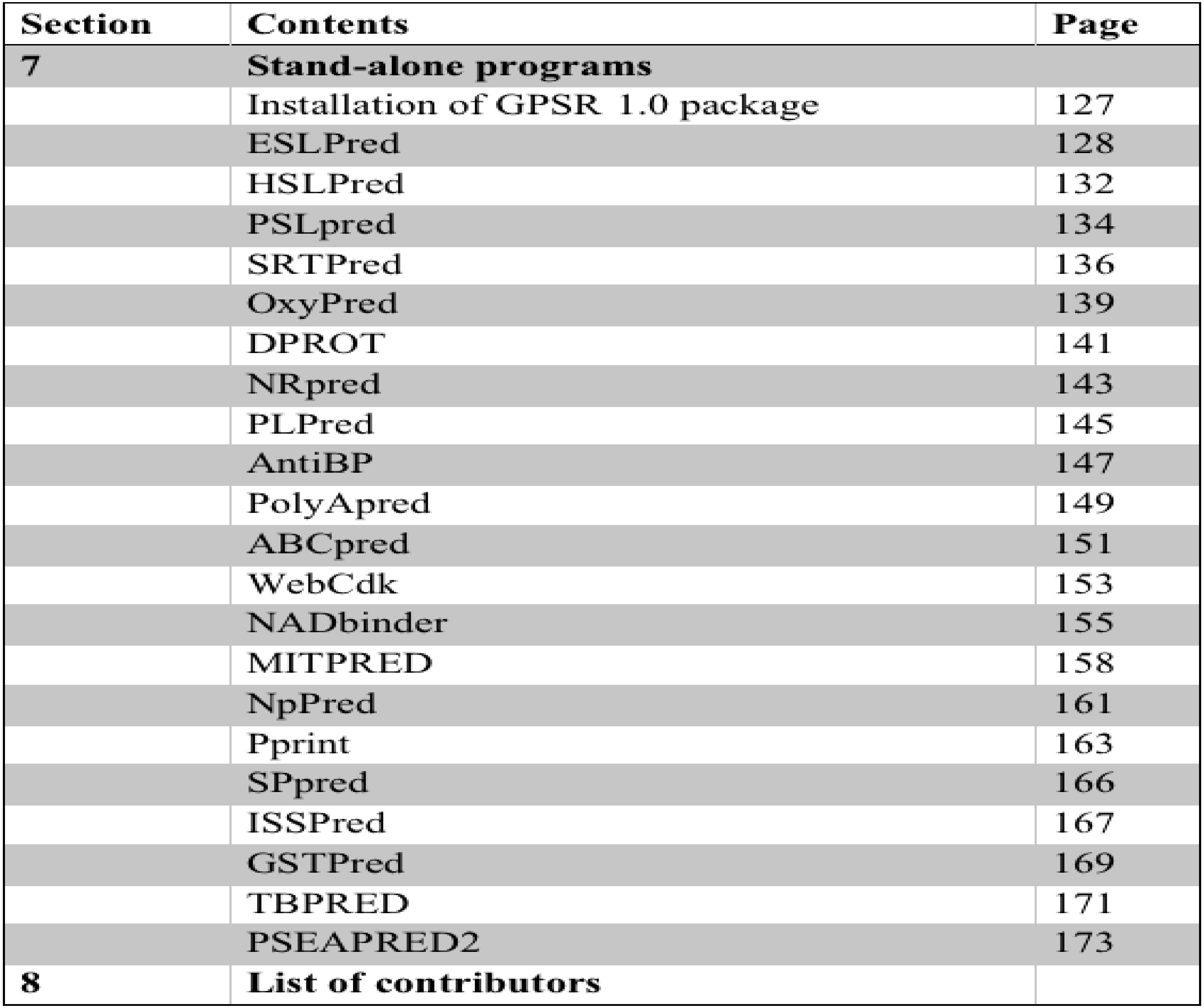
Small programs as building unit

- Why PERL?
- Why not BioPerl?
- Why not PERL modules?
- Advantage of independent programs
  - Language independent
  - Can be run independently



Program	Purpose
❖ fasta2sfasta	Convert fasta format to single fasta format
❖ pro2aac	To calculate amino acid composition of protein
❖ pro2aac_nt	To calculate amino acid composition of N-terminal (nt) residues of a protein
❖ pro2aac_ct	To calculate amino acid composition of C-terminal (ct) residues of a protein
❖ pro2aac_rest.pl	To calculate amino acid composition of a protein after removing N-, and C-terminal residues
❖ pro2aac_split	To calculate split amino acid composition (SSAC) of a protein
❖ pro2dpc	To calculate dipeptide composition of protein
❖ pro2dpc_nt	To calculate dipeptide composition of N-terminal (nt) residues of a protein
❖ pro2dpc_ct	To calculate dipeptide composition of C-terminal (ct) residues of a protein
❖ pro2tpc	To calculate tripeptide composition of protein
❖ add_cols	To add columns of two files
❖ col2svm	To generating SVM_light input format
❖ col_mult	To multiplying each column of input file with a number
❖ col_mult_sel	To multiplying selective columns with a number
❖ perl col_rem	To remove selective columns from a file
❖ col_ext	To extract selective columns from a file
❖ col_corr	To compute correlation co-efficient between two column
❖ col_avg	To calculate average column of two files
❖ seq2pssm_imp	To calculate PSSM matrix in column format without any normalization
❖ pssm_n1	To normalize pssm profile based on 1/(1+e-x) formula

Title	Description
	pro2aac (To calculate amino acid composition of protein) The amino acid composition in a protein is simply the percentage of the different amino acids represented in a particular protein. The aim of calculating the composition of proteins is to transform the variable length of protein sequences to fixed length feature vectors. In addition the conversion of a protein sequence to a vector of 20 dimensions using amino acid composition will encapsulate the properties of the protein into the vector. The composition of all 20 natural amino acids were calculated by using the following equation $\text{Composition of amino acid } i = \frac{\text{Total number of amino acid } i \times 100}{\text{Total number of all amino acids in protein}}$ Where i can be any amino acid
Usage	*pro2aac -i seq.sfa -o seq.out*
-i	Input file name contains single fasta format
-o	Output file name gives amino acid composition
seq.sfa	>seq_1##MRNRGFGRRELLVAMAMLVSVTGCARHASGARPASTTLPAGADLADRFAEL ERRYDARLGVYVPATGTTAAIE >seq_2##ACGRGFGVKLACNMNNACRTYFSDVAMAMLVSVTGCARHASGARPASTTL PAGADLADIEYRADERFAFCSTF
seq.out	# Amino Acid Composition of proteins # A , C , D , E , F , G , H , I , K , L , M , N , P , Q , R , S , T , V , W , Y, 19.18, 1.37, 4.11, 5.48, 2.74, 9.59, 1.37, 1.37, 0.00, 9.59, 4.11, 1.37, 4.11, 0.00,13.70, 4.11, 8.22, 6.85, 0.00, 2.74, 19.18, 6.85, 5.48, 2.74, 6.85, 8.22, 1.37, 1.37, 1.37, 5.48, 4.11, 4.11, 2.74, 0.00, 8.22, 6.85, 6.85, 5.48, 0.00, 2.74,
Vector	20 dimension (i.e 20 types of amino acid composition is generated)



Modelling of Immune System for Designing Epitope-based Vaccines

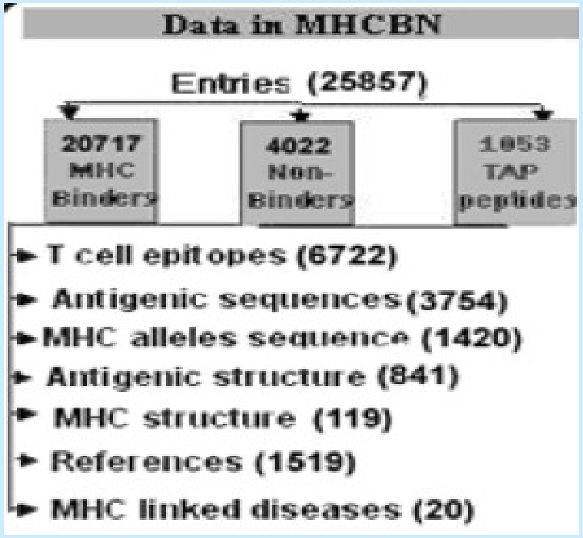
Adaptive Immunity (Cellular Response) : T_{helper} Epitopes

Propred: for promiscuous MHC II binders

MMBpred:for high affinity mutated binders

MHC2pred: SVM based method

MHCBN: A database of MHC/TAP binders and non-binders



Adaptive Immunity (Cellular Response) : CTL Epitopes

Pcleavage: for proteome cleavage sites

TAPpred: for predicting TAP binders

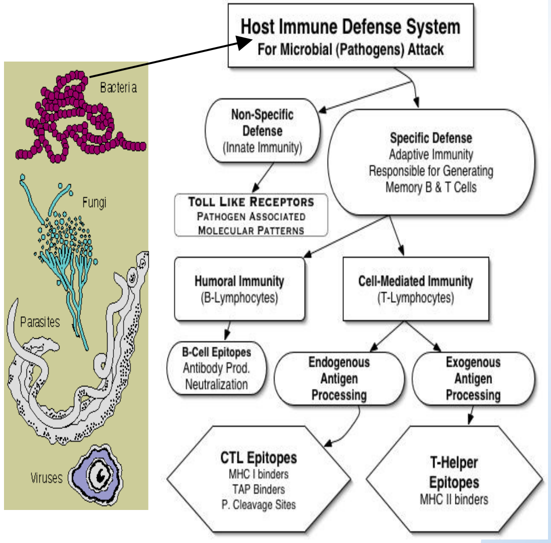
Propred1: for promiscuous MHC I binders

CTLpred: Prediction of CTL epitopes

Adaptive Immunity (Humoral Response) :B-cell Epitopes

BCIpep: A database of B-cell eptioes;
ABCpred: for predicting B-cell epitopes
ALGpred: for allergens and IgE eptopes

HaptenDB: A datbase of haptens


Host Immune Defense System
For Microbial (Pathogens) Attack
Bacteria
Fungi
Parasites
Viruses
Non-Specific Defense (Innate Immunity)
Specific Defense Adaptive Immunity Responsible for Generating Memory B & T Cells
TOLL LIKE RECEPTORS PATHOGEN ASSOCIATED MOLECULAR PATTERNS
Humoral Immunity (B-Lymphocytes)
Cell-Mediated Immunity (T-Lymphocytes)
B-Cell Epitopes Antibody Prod. Neutralization
Endogenous Antigen Processing
Exogenous Antigen Processing
CTL Epitopes MHC I binders TAP Binders P. Cleavage Sites
T-Helper Epitopes MHC II binders

Innate Immunity : Pathogen Recognizing Receptors and ligands

PRRDB: A database of PRRs & ligands

Antibp: for anti-bacterial peptides

Signal transduction in Immune System

Cytopred: for classification of Cytokines

Computer-Aided Drug Discovery

Searching Drug Targets: Bioinformatics

Genome Annotation

FTGpred: Prediction of Prokaryotic genes

EGpred: Prediction of eukaryotic genes

GeneBench: Benchmarking of gene finders

SRF: Spectral Repeat finder

Comparative genomics

GWFASTA: Genome-Wide FASTA Search

GWBLAST: Genome wide BLAST search

COPID: Composition based similarity search

LGEpred: Gene from protein sequence

Subcellular Localization Methods

PSLpred: localization of prokaryotic proteins

ESLpred: localization of Eukaryotic proteins

HSLpred: localization of Human proteins

MITpred: Prediction of Mitochndrial proteins

TBpred: Localization of mycobacterial proteins

Prediction of drugable proteins

Nrpred: Classification of nuclear receptors

GPCRpred: Prediction of G-protein-coupled receptors

GPCRsclass: Amine type of GPCR

VGIchan:_Voltage gated ion channel

Pprint: RNA interacting residues in proteins

GSTpred: Glutathione S-transferases proteins

Protein Structure Prediction

APSSP2: protein secondary structure prediction

Betatpred: Consensus method for β-turns prediction

Bteval: Benchmarking of β-turns prediction

BetaTurns: Prediction of -turn types in proteins

Turn Predictions: Prediction of α/ β/γ -turns in proteins

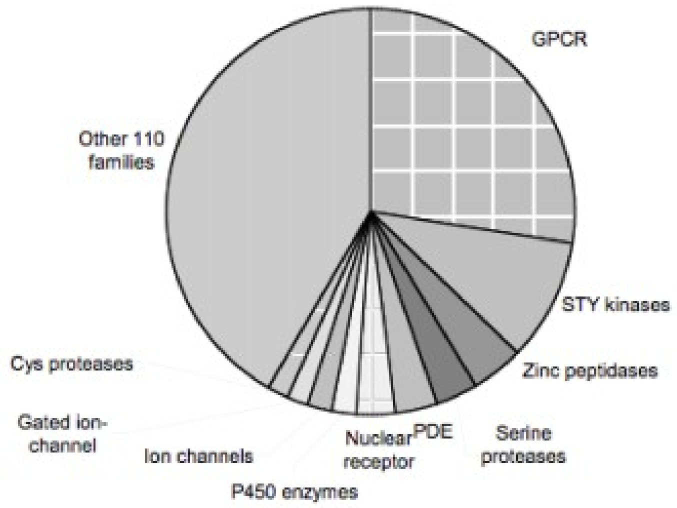
GammaPred: Prediction of-turns in proteins

BhairPred: Prediction of Beta Hairpins

TBBpred: Prediction of trans membrane beta barrel proteins

SARpred: Prediction of surface accessibility (real accessibility)

PepStr: Prediction of tertiary structure of Bioactive peptides



Thanks for Listening and Wish Collobrative Research with Russian Scientist